doi.org/10.54473/IJTRET.2025.9407
International Journal of Trendy Research in Engineering and Technology
Volume 9 Issue 4 August 2025
ISSN NO 2582-0958

LINGOBRIDGE AI: A MULTILINGUAL INTELLIGENT SYSTEM FOR TRANSLATION AND SUMMARIZATION OF INDIAN LANGUAGES

[1]Dr.Pramod Kumar Naik, [2]Divyank Jain Singhvi , [2]Ekta Ghosh, [2]Saswat.S Rout, [2]Savhar Verma

[1]Assistant Professor , [2]UG Students, Dept. of Computer Science and Engineering, School of Engineering,
Dayananda Sagar University Devarakaggalahali, Karnataka

Corresponding author: ektaghosh2002@gmail.com

Received 28 April 2025 Received in Revised form 01 May 2025 Accepted 02 May 2025

ABSTRACT

This project aims to develop an efficient multilingual translation and summarization tool tailored for Indian languages, addressing the digital communication challenges faced by speakers of low-resource languages. Existing tools often lack contextual understanding, resulting in inaccurate translations. LingoBridge AI introduces a transformer-based LLM that bridges this gap by accurately translating and summarizing documents, speech, and text. It employs language clustering and a virtual intermediary called the "Lamp Language" to reduce errors in inter-cluster translation. The model ensures real-time performance and accessibility through web and mobile platforms. Users receive results in both text and audio formats in their preferred language—Hindi or English. This system promotes digital inclusivity and empowers citizens by improving access to essential information. Ultimately, the initiative supports India's linguistic diversity while fostering an equitable and digitally connected society.

Keywords: Multilingual LLM, Translation, Indian Languages, Lamp Language, Real-time Summarization, Digital Inclusivity.

I.INTRODUCTION

The rise of multilingual large language models (LLMs) has transformed natural language processing (NLP) by giving machines the ability to understand and create text in many different languages. These advanced models can handle important language tasks, like translation, summarization across multiple languages. This technology opens up big possibilities for communication between languages and makes digital information accessible to more people.

In a country as linguistically diverse as India, with hundreds of languages, multilingual LLMs have great potential to break down language barriers. By supporting communication in regional languages, these models can help more people access technology and information, promoting digital inclusivity. This project focuses on creating a multilingual LLM system to support several regional Indian languages, including Hindi, English, Tamil, Bengali, and others, so that users can interact with technology in their preferred language across different applications.

Our project focuses on developing a multilingual transformer-based model tailored specifically for Indian languages[1] . It is designed to support Hindi, English, and a variety of regional languages to foster inclusive communication across the country's linguistically diverse population. By bridging linguistic gaps between communities, the model enhances digital participation and ensures that no one is left behind in the digital ecosystem. One of the standout features of this system is its real-time translation and summarization capabilities, which provide users with instant access to information in their preferred language. This functionality greatly improves accessibility, particularly for individuals residing in remote and underserved areas. Furthermore, the system contributes to better governance by simplifying the delivery of government content and services. Businesses can also benefit from this technology through efficient multilingual communication with clients and stakeholders[2]. By enabling seamless interaction in native languages, the platform empowers users at every level. Ultimately, the project supports the creation of a more connected, informed, and equitable digital India.

II. LITERATURE SURVEY

The Jiten Parmar, Naveen Saini, and Dhananjoy Dey proposed the ILrLSUMM model, which employs a differential evolutionary algorithm for extractive summarization of Indian regional languages, including Hindi and Gujarati. Their approach achieved notable performance improvements with a focus on optimization. This research underscores the significance of leveraging evolutionary algorithms for low-resource Indian languages and optimizing summarization processes. [3]

Nomi Baruah, Shikhar Kr. Sarma, and Surajit Borkotokey critically reviewed text summarization approaches for Indian languages, identifying challenges due to morphological complexities and a lack of annotated corpora. The study highlights the need for language-specific tools and resources to advance text summarization systems for Indian languages. [4]

Milam Aiken and Shilpa Balan conducted a comprehensive study on Google Translate (GT) accuracy by analyzing 2,550 language-pair combinations. Their research found that Western languages had better translation accuracy compared to Asian languages. BLEU scores were used to evaluate GT's performance, showing strong correlation with human assessments. The study concluded that while GT provides adequate translations for basic comprehension, it remains unreliable for complex texts. This research highlights the strengths and limitations of machine translation, emphasizing the need for human verification in critical use cases.[5]

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin proposed the Transformer model, which replaces recurrent and convolutional networks with a self-attention mechanism for sequence transduction tasks. Their approach significantly improves machine translation efficiency and accuracy while reducing training time. The Transformer model achieved state-of-the-art BLEU scores on English-German (28.4) and English-French (41.0) translation tasks. This research highlights the impact of self-attention in enabling parallelization, reducing computational costs, and enhancing model performance, laying the foundation for modern NLP architectures like BERT and GPT.[6]

III. OBJECTIVES

The primary objective of the LingoBridge AI project is to develop an efficient multilingual translation and summarization system tailored for Indian languages, with a special focus on Hindi and English. This system aims to overcome communication barriers by leveraging a transformer-based Large Language Model (LLM) capable of handling text, speech, and document inputs. A unique feature of this model is the introduction of a virtual intermediary, known as the Lamp Language, which helps maintain contextual accuracy and minimize semantic drift, especially during inter-cluster translations. To optimize the model’s performance, linguistically similar languages are grouped into clusters, enabling better learning and reduced translation errors. The project further aims to support real-time translation and summarization through responsive mobile and web applications, ensuring broad accessibility. Additionally, LingoBridge is designed to empower users by facilitating access to government, educational, and job-related content in their native or preferred language. It contributes to digital inclusivity while preserving linguistic diversity, and its scalable architecture allows for the future integration of more Indian languages and API support across platforms.

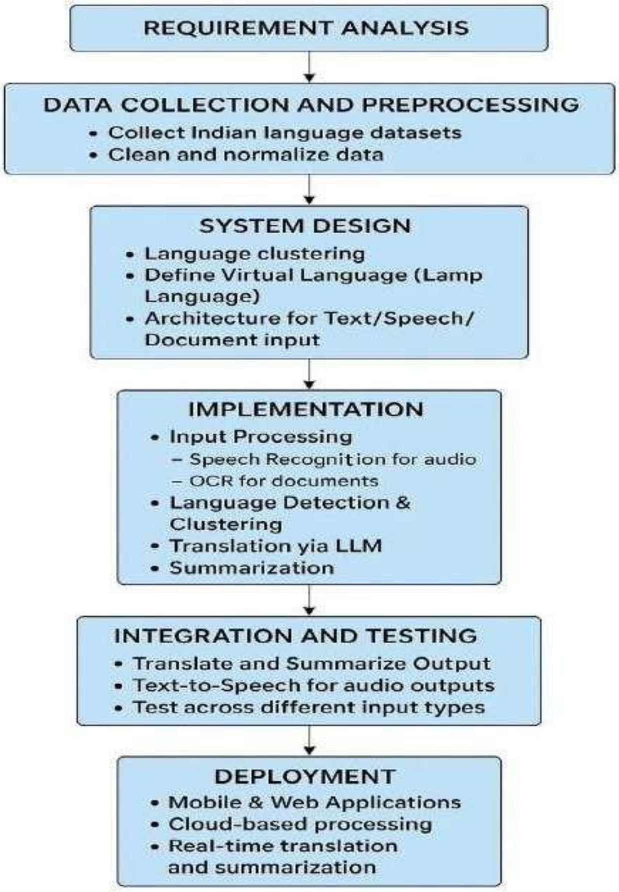
IV. METHODOLOGY

The methodology of our multilingual translation and summarization system follows a structured approach to ensure accuracy, efficiency, and inclusivity. It begins with data collection and preprocessing, where Indian language datasets, including low-resource languages, are gathered. The data is then cleaned and normalized to remove noise, inconsistencies, and formatting issues. To enhance translation accuracy, language clusters are created based on linguistic similarities, minimizing errors in cross-language communication. The next phase involves input processing, where the system categorizes user input as text, speech, or document. If the input is speech, a Speech Recognition Module converts it into text while filtering out background noise. For document- based inputs, Optical Character Recognition (OCR) and NLP techniques extract and process text efficiently. Once the text is obtained, it undergoes language detection and clustering using NLP models, which map the detected language to predefined language clusters. In cases where direct translation is complex, a Virtual Language (Lamp Language) serves as an intermediary layer to preserve contextual meaning and minimize translation errors.

The LLM-based translation and summarization process ensures that the extracted text is accurately translated while maintaining the original context. A summarization algorithm further refines long documents or conversations into concise outputs, making information more digestible. The final translated and summarized text is then presented in the user’s selected language (Hindi or English). If audio output is preferred, a Text-to-Speech (TTS) system converts the translated text into natural, human-like speech for better accessibility.

To enhance performance, the system incorporates continuous learning and model refinement. It collects user feedback and interaction data to fine-tune the model, ensuring it adapts over time. This ongoing improvement helps bridge contextual gaps, refine translations, and expand language support. Additionally, the dataset is periodically updated to incorporate regional dialects and evolving linguistic patterns. Finally, the deployment and accessibility phase ensures that the model is available as a mobile and web application for real-time language translation and summarization.

The platform supports both text and voice-based interactions, making it accessible to a diverse user base across India. With cloud-based processing and scalable architecture, the system efficiently handles large volumes of data while maintaining speed and accuracy. By integrating LLM, NLP, and Virtual Language clustering, this methodology effectively reduces language barriers, fostering seamless communication across industries, government sectors, and diverse linguistic communities. It promotes inclusivity, accessibility, and digital empowerment, ensuring that language is no longer a barrier to participation in the digital world.


REQUIREMENT ANALYSIS
DATA COLLECTION AND PREPROCESSING
• Collect Indian language datasets
• Clean and normalize data
SYSTEM DESIGN
• Language clustering
• Define Virtual Language (Lamp Language)
• Architecture for Text/Speech/ Document input
IMPLEMENTATION
• Input Processing
– Speech Recognition for audio
– OCR for documents
• Language Detection & Clustering
• Translation yia LLM
• Summarization
INTEGRATION AND TESTING
• Translate and Summarize Output
• Text-to-Speech for audio outputs
• Test across different input types
DEPLOYMENT
• Mobile & Web Applications
• Cloud-based processing
• Real-time translation and summarization

V. MODEL ARCHITECTURE

The core of the LingoBridge AI system is built upon a transformer-based multilingual Large Language Model (LLM)[1]. This model follows the encoder-decoder structure, which allows it to handle both translation and summarization tasks effectively. The input sequence is first passed through an embedding layer, followed by positional encoding to maintain the order of tokens. The encoder processes the input using multi-head self- attention and feed-forward layers, capturing long-range dependencies and semantic meaning.

The decoder, equipped with masked multi-head attention, generates the translated or summarized output while attending to relevant parts of the encoded input. Each block includes layer normalization, residual connections, and dropout to ensure model stability and generalization. A virtual intermediary language, referred to as the Lamp Language, is incorporated during training for cross-cluster translations. This intermediary representation improves translation fidelity between linguistically distant language pairs by preserving contextual information.

The model is trained on over one million parallel Hindi-English and regional language sentence pairs using supervised learning. A backtracking mechanism is integrated to allow for output correction by comparing against known results, refining performance iteratively.

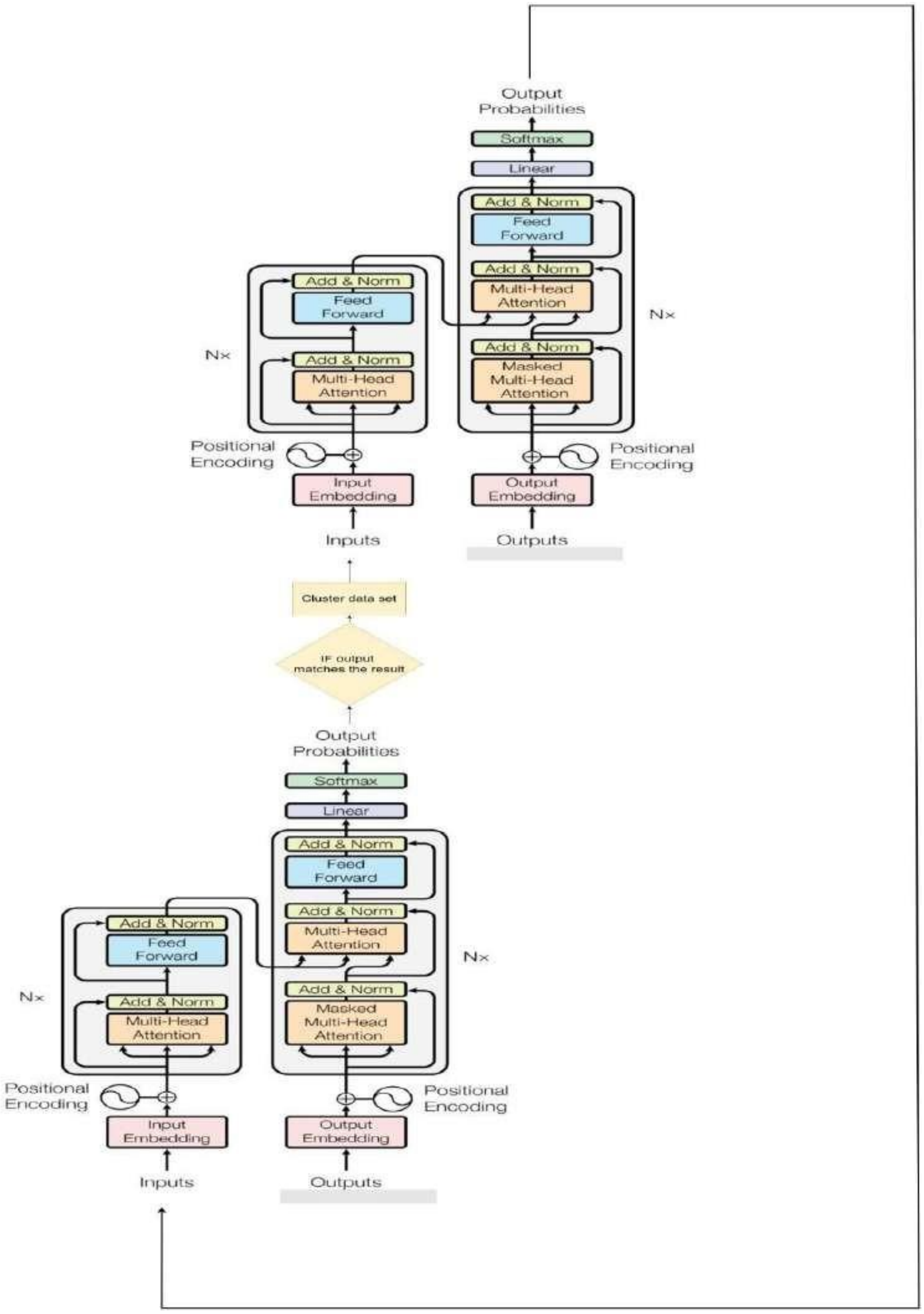


Fig-1-Transformer-Based Multilingual Model with Clustering Mechanism[6]

VI. SYSTEM ARCHITECTURE

The system architecture of LingoBridge AI is designed to support seamless multilingual translation and summarization across Indian languages. It follows a modular and scalable structure based on transformer-based neural network models. The core architecture includes components such as Input Handling, Language Detection, Language Clustering, the Transformer LLM, Lamp Language Integration, the Summarization Engine, and Output Delivery.

The architecture starts with an Input Handler that accepts data in the form of text, audio, or documents. Audio input is converted to text using a Speech-to-Text (STT) module, and documents are processed using OCR to extract text. Once the text is obtained, the Language Detection module identifies the input language.

Next, the identified language is checked against predefined Language Clusters, grouping similar languages together based on grammar, syntax, and linguistic patterns. If the source and target languages belong to different clusters, a Virtual Intermediary Language, known as the Lamp Language, is used to bridge the semantic gap, ensuring better accuracy and meaning preservation.

The core transformer-based LLM then processes the clustered and converted input, performing both translation and summarization. A Backtracking Mechanism is embedded to evaluate the output for correctness and refine the process iteratively if needed. This continuous feedback loop improves translation quality over time.

The processed output is then passed to the Output Module, where it is delivered to the user in their selected language, either in text or through Text-to-Speech (TTS)[7] conversion for audio output. The system supports real- time API interaction, enabling mobile and web applications to access translation services with minimal latency. The complete architecture is cloud-enabled, ensuring scalability, high performance, and the ability to serve multiple users simultaneously. This structure not only ensures modularity and performance but also supports continuous expansion to more Indian languages and dialects.

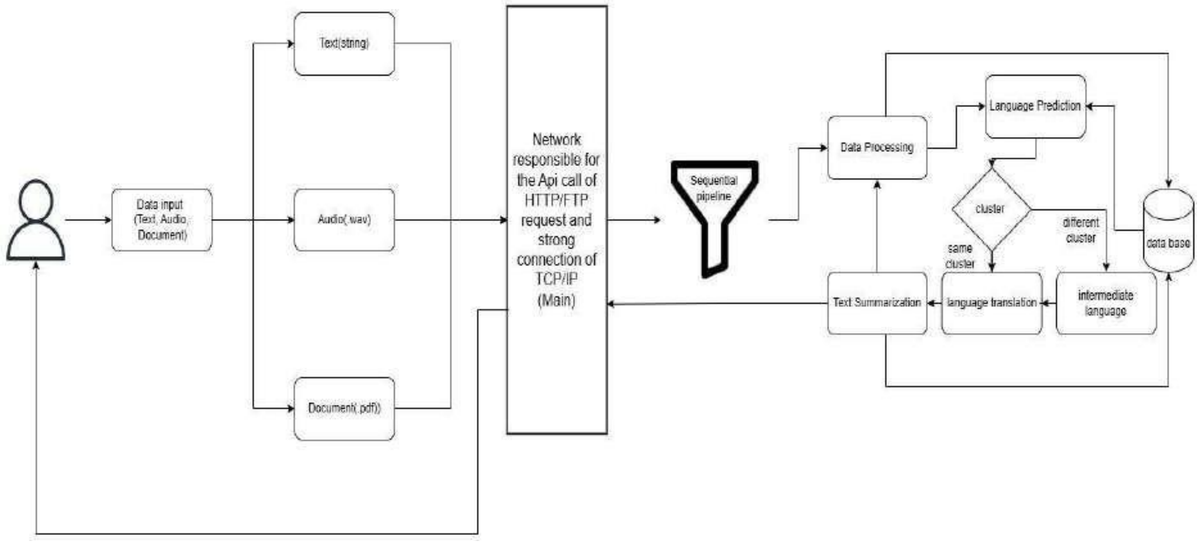


Fig- 2- System Architecture

VII. FLOW DIAGRAM AND WORKING OF THE SYSTEM

The diagram represents the workflow of a multilingual translation and summarization system using a Large Language Model (LLM). The process begins with client interaction, where users provide input in the form of text, audio, or document files. This input is sent to the server via an API call over a stable TCP/IP network connection. Once processed, the translated and summarized result is delivered back to the user. The input processing server script first determines the type of input—text, audio, or document. If the input is audio, it undergoes speech recognition to convert it into text, while background noise is removed. If it is a document, the system converts it into text using a Python script. Unsupported formats are rejected, and the extracted text is then passed into a queue for further processing.

In the model training and prediction phase, the system detects the language of the input using NLP techniques. If a trained model is available, the data is formatted for processing. Otherwise, relevant training data is extracted from the database. The LLM then translates and summarizes the text while ensuring contextual accuracy. The final output is presented in the user's desired language, either in text or speech format. The database management component plays a crucial role in storing language datasets, including previous translations and user inputs. The system continuously updates and fine- tunes the model based on new inputs, improving translation accuracy over time. This entire workflow ensures efficient, real-time multilingual translation and summarization, enabling seamless communication across different Indian languages.

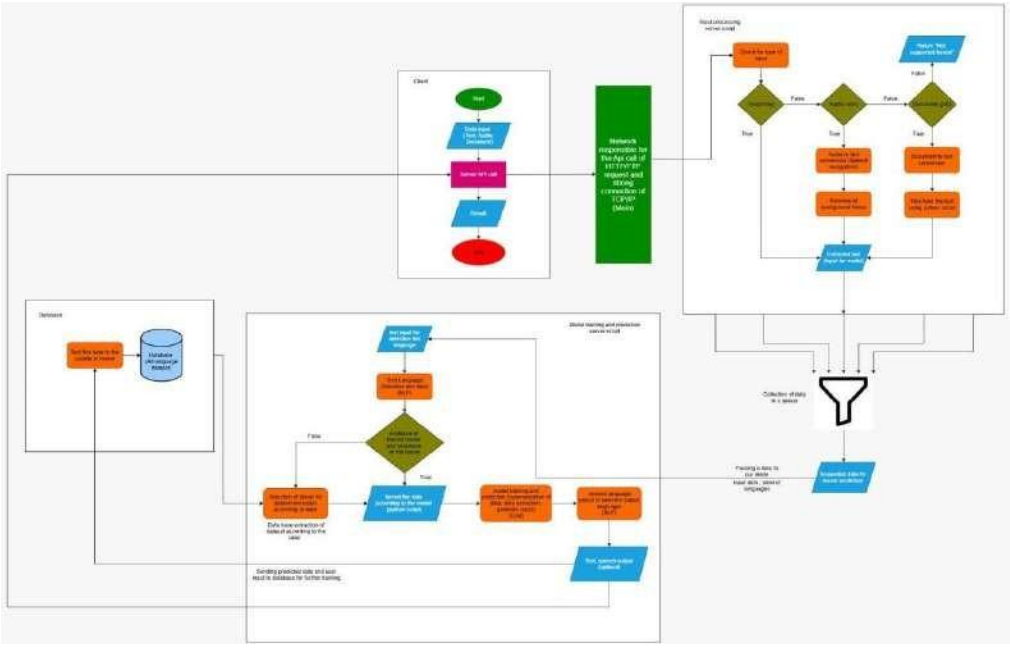


Fig-3- Flow Diagram

Data Collection:

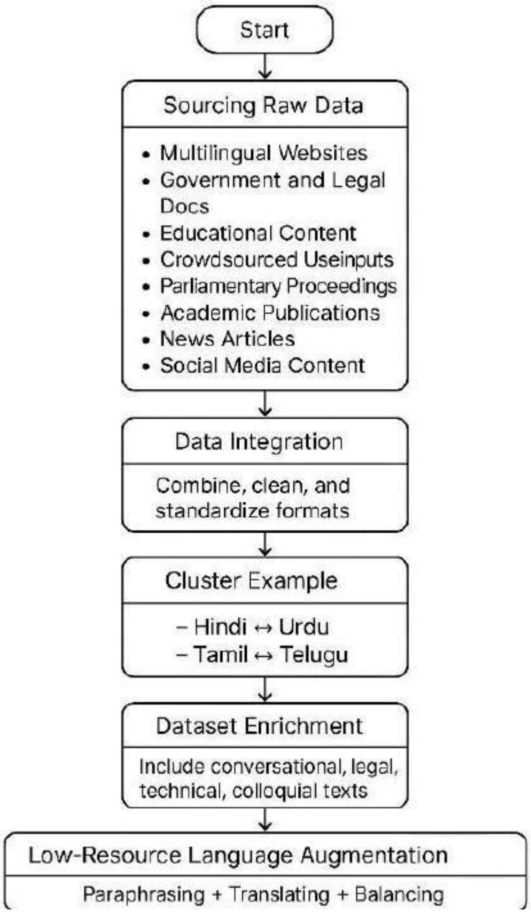


Fig-4- Data Collection

Data collection is a foundational step in the LingoBridge project, aimed at building a rich, diverse, and representative dataset that reflects the linguistic complexity of Indian languages. This process involves sourcing raw linguistic data from multiple avenues, including multilingual websites, open-access governmental and legal documents, educational content, and crowdsourced user inputs. Public domain datasets such as parliamentary proceedings, academic publications, news articles, and social media content are also leveraged to ensure the dataset captures both formal and informal language usage. By integrating these various types of content, LingoBridge ensures that the language model is trained on authentic, real-world linguistic patterns, enhancing its contextual understanding and practical translation capabilities.

A key aspect of the data collection strategy is the clustering of languages based on their grammatical structure, vocabulary overlap, and cultural context. Languages with similar roots or syntactic patterns—such as Hindi and Urdu, or Tamil and Telugu—are grouped into clusters to optimize the training process. This clustering allows the model to generalize better across similar languages while maintaining precision in language-specific nuances. Each cluster is enriched with texts that reflect a variety of use cases, such as conversational language, legal terminology, technical documents, and colloquial phrases. Additionally, low-resource languages receive special attention by augmenting the dataset with paraphrased and translated content to balance the distribution. This structured and inclusive approach to data collection plays a critical role in making LingoBridge a powerful and culturally aware multilingual translation system, capable of serving diverse Indian linguistic communities.

Data Pre-Processing:
Here's a breakdown of each step involved in the data preprocessing:
Special Character Removal:
This step involves using regular expressions (regex) to remove any unnecessary special characters or symbols from the text, such as punctuation marks, numbers, or other non-alphabetic characters. The purpose is to ensure that the data only contains meaningful text, eliminating noise that could negatively affect model performance.
Tokenizer:
The tokenizer is responsible for breaking the input text into smaller units, usually words or characters, based on predefined rules. In character-level tokenization, each character, including spaces, is treated as a token. This step converts the raw text into tokens that can be further processed by the model. Tokenizers often convert the tokens into numerical values (IDs) corresponding to each unique token in the dataset. Labeling Each Character with an ID: During this step, each character or token in the dataset is assigned a unique identifier (ID). This is done to convert text into a numerical form that can be processed by machine learning models[8]. A dictionary is typically created where each character or token is mapped to a specific ID. These IDs are used in the model's training phase, allowing the model to learn from the encoded representations.
Prediction:
In this step, the trained model takes the input data (now represented as tokens with IDs) and makes a prediction based on learned patterns. The model outputs its predicted results, which could be in the form of a classification, a sequence, or other types of output depending on the task. The prediction process involves using the trained weights and architecture to analyze the input data and provide the most likely or appropriate result.

VIII. RESULTS AND DISCUSSION

The LingoBridge system demonstrates a robust and effective approach to multilingual translation, especially between Hindi and English. The innovative clustering approach used in model training effectively minimizes translation errors by grouping linguistically similar languages. This not only improves translation fidelity within clusters but also enhances computational efficiency. One of the standout features of the system is the integration of a virtual "Lamp Language", which serves as an intermediary during inter-cluster translations. This technique significantly reduces contextual loss and semantic drift, ensuring more accurate and meaningful translations.
The incorporation of backtracking mechanisms enables continuous refinement of translation outputs, progressively improving model performance over time. This dynamic learning approach ensures the model becomes more accurate with real-world use. The system has shown high efficiency in translating official documents, legal texts, and government communications, making it suitable for real-life, high-importance applications.

Moreover, real-time implementation across both mobile and web platforms makes the system highly accessible to users from diverse backgrounds. The user-friendly interface, built using React, ensures a seamless experience for individuals, businesses, and public institutions. Low latency, high accuracy, and scalability make LingoBridge a valuable tool in India’s multilingual digital landscape. The model stands as a pioneering step toward inclusive and efficient communication in India’s complex linguistic environment.

IX. CONCLUSION

The LingoBridge AI system presents a significant advancement in real-time multilingual translation for Indian languages. The project's primary success lies in effectively bridging communication gaps between Hindi and English speakers. By reducing language bias during model training, the system delivers more inclusive and balanced translations. Its cluster-based training strategy and the innovative Lamp Language intermediary improve contextual accuracy. The adaptability of the training process to various transformer architectures enhances scalability. Real-time API integration across platforms ensures minimal latency and seamless user experiences. The system also features a responsive React-based web interface for improved accessibility. This enables individuals, businesses, and institutions to easily utilize translation services. It promotes digital inclusivity, especially for underserved and linguistically diverse communities.

The dual availability on mobile and web platforms ensures wide accessibility and usability. LingoBridge supports both text and audio inputs, making it versatile across use cases. User feedback has been positive, validating the system's accuracy and reliability. The model is designed for continuous learning and performance improvement over time. Future work includes expanding support to more Indian languages and dialects. Overall, LingoBridge lays the foundation for an inclusive, multilingual digital ecosystem in India.

ACKNOWLEDGEMENT

Authors Divyank Jain Singhvi ,Ekta Ghosh , Saswat.S. Rout ,and Savhar Verma owe a special debt of gratitude to their project guide, Dr. Pramod Kumar Naik, Associate Professor, Department of Computer Science and Engineering, whose expertise, dedication, and mentorship played a critical role in the successful completion of this project. His timely suggestions, detailed reviews, and constant support helped us overcome challenges and refine our research.

REFERENCES

[1].V. Desai, S. Patel, And M. Thomas, “Code-Mixed Hinglish To English Language Translation Using Transformer Models,” In Proc. 2024 IEEE Int. Conf. On Computational Linguistics (Coling), 2024.

[2]. P. Kumar, R. Singh, And A. Gupta, “A Real-Time End-To-End Multilingual Speech Recognition System,” In Proc. IEEE Int. Conf. On Signal And Information Processing (ICSIP), 2023.

[3]. J. Parmar, N. Saini and D. Dey, "An Unsupervised Evolutionary Approach for Indian Regional Language Summarization," 2024 IEEE Congress on Evolutionary Computation (CEC), Yokohama, Japan, 2024, pp. 1-8, doi: 10.1109/CEC60901.2024.10612059.

[4].N. Baruah, S. K. Sarma and S. Borkotokey, "Text Summarization in Indian Languages: A Critical Review," 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP), Gangtok, India, 2019, pp. 1-6, doi: 10.1109/ICACCP.2019.8882968.

[5].Milam Aiken and Shilpa Balan,An Analysis of Google Translate Accuracy, Translation Journal, 2011;16(2)
https://translationjournal.net/journal/56google.htm

[6].Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, Illia Polosukhin,’ Attention is All you Need’ in book titled' Advances in Neural Information Processing Systems', Curran Associates, nc.
https://proceedings.neurips.cc/paper_files/paper/201 7/ file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[7]. R. Chandra, S. Pandey, And D. Sharma, “A Unified Framework For Multilingual Text-To-Speech Synthesis,” In Proc. 2023 Int. Conf. On Natural Language Processing (Icon), 2023.

[8]. A. Nair, R. Rao, And K. Iyer, “Identification Of Top-3 Spoken Indian Languages Using Machine Learning Techniques,” In Proc. 2023 IEEE Int. Conf. On Speech Technology (Icst), Pp. 55-60, 2023.